SOLIDCROWD - CROWD FUNDING PLATFORM USING BLOCKCHAIN

A PROJECT REPORT

Submitted by

ADHARSH H (TKM21MCA-2002)

to

The APJ Abdul Kalam Technological University

In partial fulfillment of the requirements for the award of the degree of

MASTER OF COMPUTER APPLICATIONS

Thangal Kunju Musaliar College of Engineering
Kerala

DEPARTMENT OF COMPUTER APPLICATIONS

MAY 2023

DECLARATION

I undersigned hereby declare that the project report on **SOLIDCROWD - CROWD FUNDING PLATFORM USING BLOCKCHAIN**, submitted for partial fulfillment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of Dr. Fousia M Shamsudeen. This submission represents my ideas in my own words and where ideas or words of others have been included,I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not previously served as the basis for the award of any degree, diploma, or similar title by any other University.

ADHARSH H

Kollam

DEPT. OF COMPUTER APPLICATIONS

TKM COLLEGE OF ENGINEERING KOLLAM

2021 - 23

CERTIFICATE

This is to certify that the report entitled **SOLIDCROWD - CROWD FUNDING PLATFORM USING BLOCKCHAIN** submitted by **ADHARSH H** (TKM21MCA2002) to the APJ Abdul Kalam Technological University in partial fulfillment of the Masters degree in Computer Applications is a bonafide record of the project work carried out by him under our guidance and supervision. This report, in any form, has not been submitted to any other University or Institute for any reason.

Internal Supervisor Head of the Department External Examiner

ACKNOWLEDGEMENT

First and Foremost I owe sincere gratitude and heart full thanks to everyone who shared their precious time and knowledge for the successful completion of my project.

I am extremely grateful to **Dr. Fousia M Shamsudeen**, Head of the Department and project guide, Department of Computer Applications, for providing me with best facilities.

I extend profound thanks to **Prof. Vaheetha Salam** , the project coordinator, for their exceptional support and coordination.

My advisor **Prof. Natheera Beevi M** offered expert advice and valuable suggestions, enhancing the quality of my work.

I profusely thank all other faculty members in the department and all other members of TKM College of Engineering, for their guidance and inspirations throughout my course of study.

I owe my thanks to my friends and all others who have directly or indirectly helped me in the successful completion of this project.

ADHARSH H

ABSTRACT

SOLIDCROWD - CROWD FUNDING PLATFORM USING BLOCKCHAIN, aims to leverage blockchain technology to create a new decentralized crowdfunding platform that removes intermediaries and enhances transparency and security. The platform will allow project creators to directly interact with backers, providing a transparent and secure environment for crowdfunding campaigns. By leveraging blockchain, the proposed crowdfunding platform eliminates the need for intermediaries, allowing direct interaction between project creators and backers. Smart contracts deployed on the blockchain ensure the transparent and immutable execution of crowdfunding campaigns. The decentralized nature of the blockchain network enhances security, mitigating the risk of fraudulent activities and increasing trust among participants. The project also aims to address the issue of financial inclusion by allowing backers to participate using various cryptocurrencies, thereby removing barriers associated with traditional banking systems. Smart contract functionality ensures automated and secure distribution of rewards or returns to project backers, reducing administrative overheads and increasing efficiency..

Contents

List of Figures

Chapter 1

Introduction

SOLIDCROWD - CROWD FUNDING PLATFORM USING BLOCKCHAIN is a decentralized crowd funding platform that works on the Ethereum blockchain network. However, traditional crowdfunding platforms have several limitations, including intermediaries, lack of transparency, and potential risks of fraud. The emergence of blockchain technology offers a promising solution to these challenges, by enabling a decentralized and transparent approach to crowdfunding.

This project aims to leverage blockchain technology to create a new decentralized crowdfunding platform that removes intermediaries and enhances transparency and security. The platform will allow project creators to directly interact with backers, providing a transparent and secure environment for crowdfunding campaigns.The integration of smart contracts on the blockchain will ensure the transparent and immutable execution of crowdfunding campaigns. These self-executing contracts enforce the terms and conditions of the campaign, eliminating the possibility of manipulation or fraud. Backers can have confidence that their contributions will be used as intended, and project creators can focus on realizing their ideas without concerns about payment disputes or misappropriation of funds.

The decentralized nature of the platform will break down geographical barriers, enabling global participation. Entrepreneurs and innovators from around the world can present their projects to a diverse and international community of backers. This fosters cross-border collaboration, expands networking opportunities, and unlocks a larger pool of potential supporters.

1.1 Problem Statement

This project aims to utilize blockchain technology in the crowdfunding process. By leveraging blockchain, the project seeks to provide a decentralized crowdfunding platform that resolves the above-mentioned problems. The platform will offer transparency, eliminating the need for intermediaries and ensuring the efficient and secure utilization of funds. It will also enhance trust by providing immutability and transparency of transactions.

By incorporating blockchain's features, such as smart contracts, the project aims to automate processes, ensure accountability, and prevent fraudulent activities. Additionally, the utilization of cryptocurrencies and blockchain's decentralized nature will promote financial inclusion, allowing individuals worldwide to participate without relying on traditional banking systems. The elimination of geographical barriers will enable global collaboration, expanding networking opportunities for project creators and providing access to a diverse pool of potential backers.

1.2 Objectives

- Develop a decentralized crowdfunding platform
- Ensure transparency and accountability
- Enhance security and mitigate fraud risks
- Enable global participation and collaboration

1.3 Existing System

Traditional crowdfunding platforms are centralized and rely on third-party payment processors. They have control over the funds raised, the campaigns that are allowed on the platform, and the fees that are charged to users. This lack of transparency and control can make it difficult for project creators to raise funds and for backers to trust the platform. Additionally, traditional crowdfunding platforms often rely on third-party payment processors, which can add additional fees and delays to the fundraising process. Few established crowdfunding platforms such as Kickstarter and Indiegogo have revolutionized the start-up world with the flexibility and efficiency in raising funds. Blockchain based crowdfunding might be the next step in evolution of fundraising platforms assisting start up founders in the journey of building their dream idea. The major issues with these established crowdfunding platforms are that they are centralized bodies controlled by a corporation charging high fees and influencing campaigns. Blockchain based crowdfunding platform can help this process by decentralizing the funding model from the likes of Kickstarter and other companies. Blockchain's distributed ledger helps in getting rid of the centralized intermediaries such as Kickstarter and Indiegogo that take huge amounts of money from a campaign as a maintenance fee. Blockchain crowdfunding is a purer form of crowdfunding as it removes any intermediaries between the backers and the start-up.

1.4 Limitations of Existing System

- Exorbitant fees: Usually, crowdfunding platforms take a certain amount of fee for every project that is listed. Sometimes, it's a specific amount, and other times, it is taken as a percentage of the contribution made by the contributors. This is a drawback for the availability of the funds since start-ups are literally looking for every rupee to help themselves.

- IP risk: sometimes, few start-ups don't protect their intellectual property and in order expose them to other experienced investors who can steal their idea and enter the market with the resources they could arrange for.

- DIY marketing: sometimes, its aggressive advertising and marketing that the start-ups need, rather than spreading the word. And this kind of marketing requires a huge expenditure which again falls as a challenge on the funds for the start-ups.

- Fine print rules and regulations: not that all platforms accept all kinds of services to be provided for. They have their own criterion, and if the start-up isn't meeting the criteria mentioned, then it poses as an obstacle for a start-up's innovation and business.

1.5 Proposed System

This system is aimed to overcome the above major shortcomings with current crowd funding platforms. Crowd fundraising involves a large number of transactions, it is necessary to manage and document them legally. As a result, a smart contract is utilized, which is a transaction protocol that automatically executes, controls, and documents transactions on behalf of project creators and investors in accordance with the agreement. Any web-based application is a centralized application which means that everything done on the platform is controlled by a single company server. Decentralized application is offered based on the Ethereum Blockchain, in which all campaign information, contributions,withdrawal requests, and funds are stored on a blockchain network that is open to all. The concept is called "Distributed ledger technology.

The distributed ledger and its contents are available to all network participants. Here, the transaction uses PoS in which it is more fast and secure than existing PoW. It is energy efficient in which the nodes are not competing against each other to attach a new block to the blockchain, energy is saved. Also, no problem has to be solved (as in case of Proof-of-Work system) thus saving energy. Proof of stake cuts out the need for complex computations. So, it beats proof of work when it comes to energy efficiency. A transactional record that cannot be changed. Transactions are recorded only once with this shared ledger, reducing the duplication of effort. After a transaction is recorded to the shared ledger, no participant can edit or tamper with it. If a mistake is found in a transaction record, a new transaction must be entered to correct the problem, and both transactions are then visible. This means that all nodes on the blockchain may see and store funds and transactions preventing data from being held on a single or centralized server. As a result, safeguarding the funds from falling into the wrong hands and being misused is an elegant and practical solution to the situation at hand.

1.6 Feasibility Study

A feasibility study is a test of system proposal according to its workability. Impact on the organization, ability to meet user needs and effective use of resources. The objective of feasibility study is not to solve the problem, but to acquire a sense of its scope. During the study, the problem definition is crystallized and aspects of the problem to be included in the system are determined, consequently costs and benefits are estimated with greater detail at this stage. Feasibility Analysis involves eight steps: From a project team and appoint a project leader, prepare a system flow chart, enumerate potential candidate systems, Describe and identify characteristics of candidate systems, Describe and evaluate performance and cost effectiveness of each candidate systems, weight system performance and cost data, select the best candidate system and prepare and report final project directive and management. Five key considerations involved in the feasibility analysis:

- Economic feasibility
- Operational feasibility
- Technical feasibility
- Behavioural Feasibility
- Legal Feasibility

1.6.1 Economic Feasibility

This study present tangible and intangible benefits from the project by comparing the development and operational cost. Economic feasibility is a crucial consideration for any project, especially for a blockchain-based crowdfunding platform. Such platforms are designed to bring together entrepreneurs and investors on a decentralized, transparent and secure platform, thereby reducing the friction and costs involved in traditional fundraising methods. The economic feasibility of a blockchain-based crowdfunding platform is determined by several factors, including the size of the market, the level of competition, and the cost of developing and maintaining the platform. The blockchain technology used in such a platform has the potential to revolutionize the way we think about fundraising, but it also requires careful consideration of its economics. One of the primary revenue streams for a blockchain-based

crowdfunding platform is the transaction fee charged to campaign creators for each contribution they receive. This fee is generally lower than traditional fundraising fees, but the volume of contributions required to generate significant revenue must be considered. The platform's cost structure includes the cost of developing and maintaining the blockchain infrastructure, legal compliance, marketing, and network security. These costs must be balanced against the revenue generated by the platform to ensure economic viability.

1.6.2 Operational Feasibility

Study conducted in 15 subjects, all of them are convinced and consented for the assessment. Since study is new to all learners in their comfort zone, they showed interested of the learning. After the assessment of each simulation are so designed to conduct that it will drag the interest of the learning. After the assessment of each simulation and automatic prediction of the preferred learning style the system satisfies the user need by providing more reliable and biologically convinced solution. This system should easy to operate by the user and it should also be flexible to the requirements. Proposed systems are beneficial only if they can be turned into information system that will meet the organization operating requirements. People are inheritably resistant to change and computers have been known to facilitate change. In this project system man power and time factors are reduced.

1.6.3 Technical Feasibility

The use of React js, Solidity, Node js, and Hardhat in the development of the blockchain-based crowdfunding platform demonstrates its technical feasibility. Hardhat is a development environment that helps developers build, test, and deploy smart contracts securely and efficiently. By using Hardhat, we ensure that smart contracts are error-free, optimized for gas usage, and adhere to industry-standard security best practices, which enhances the overall technical feasibility and security of the platform. React js provides a powerful user interface, while Solidity enables the creation of smart contracts and Node js facilitates server-side scripting, all of which are essential components of a blockchain-based platform. Additionally, the use of these technologies helps ensure compatibility with existing systems and enhances the platform's scalability, data security, and privacy. One of the primary benefits of using blockchain technology in a crowdfunding platform is the increased level of data security and

privacy. The platform must be designed to ensure that user data and transaction information are protected against cyber-attacks and other forms of data breaches. It should also ensure that all user data is kept private and confidential. The technical feasibility of the proposed blockchain-based crowdfunding platform is designed to enhance user experience and user engagement. By utilizing React js, the platform offers a highly intuitive and interactive user interface, making it easy for campaign creators and backers to navigate the platform and engage with campaigns. The use of Solidity and Node js enables the platform to handle a high volume of transactions efficiently, ensuring that users can contribute to campaigns without experiencing delays or technical issues. The platform's compatibility with existing payment systems further enhances the user experience, allowing for seamless and convenient contributions to campaigns. Overall, the technical feasibility of the platform is designed to provide users with a seamless and engaging experience, promoting increased user engagement and revenue growth.

1.6.4 Behavioural Feasibility

Studied whether the performance of proposed blockchain-based crowdfunding system meet proposal expectation. Under this feasibility study pointed designed system is worth to small capital fund raising sectors. Since the blockchain is an emerging technology in world, previous studies achieved output. The blockchain-based nature of the platform provides a high level of transparency and security, which will communicate to users to build trust in the platform. Additionally, This will ensure that campaign creators provide clear and concise information about their projects, including their funding goals, milestones, and expected outcomes, to enable users to make informed decisions about their contributions. The platform's success is highly dependent on user adoption and engagement, which requires an in-depth understanding of user behaviour and preferences. This feasibility study is carried out by small group of people who are familiar with information testing techniques, who understand the parts of the problem of existing system that are relevant to the project and are skilled in analysis and design process.

1.6.5 Legal Feasibility

To ensure legal feasibility, legal experts will be consulted to ensure that the platform follows all laws and regulations concerning securities and crowdfunding. The platform will also implement measures to verify all campaigns and contributors to prevent fraudulent activities

and comply with anti-money laundering and know-your-customer regulations. The platform will also ensure that all campaign creators provide accurate and complete information about their projects to prevent misleading investors and comply with consumer protection laws. Additionally, all campaigns will be required to be registered and approved by relevant regulatory bodies before being listed on the platform. The platform will also incorporate smart contracts to ensure that all transactions are transparent, secure, and enforceable, which will mitigate legal risks associated with traditional contract disputes. Lastly, the platform will be regularly monitored and assessed for legal compliance, and any necessary updates will be made to reflect changes in laws and regulations.

Chapter 2

Literature Survey

A literature review is a comprehensive analysis and interpretation of the relevant literature on a particular subject. Using a literature review, research questions are formulated, and then answers are sought by searching for and analysing pertinent literature. The re-analysis of the study's results frequently yields new insights, which is an advantage of literature reviews.A literature review is both a summary and an explanation of the complete and current state of knowledge on a subject, as presented in academic books and journal articles. There are two types of literature reviews that you may be asked to write in college: one is written as an independent assignment during a course.The second is an introduction or preparatory work for a longer piece, such as a thesis or research report. The type of review you are writing will determine the review's focus, perspective, and the formulation of a clear hypothesis or thesis argument. By reading published literature reviews or the introductory chapters of relevant theses and dissertations, you can learn the distinctions between these two types. Consider the structure of their arguments and how they approach the issues.

2.1 Purpose of the Literature Review

1. It makes research on a specific topic accessible to readers by selecting and summarising high-quality articles or studies that are relevant, meaningful, significant, and valid.

2. It offers beginning researchers in a new field an excellent starting point by requiring them to summarise, evaluate, and compare original research in that field.

3. It ensures that researchers do not repeat work already completed.

4. It may provide hints as to the direction of future research or suggest areas of emphasis.

5. It emphasises the principal findings.

6. It identifies contradictions, gaps,contradictions, and inconsistencies in the literature.

7. It provides a constructive analysis of other researchers' methodologies and approaches.

2.2 Related Works

Implementing a crowdfunding project using blockchain technology is crucial for several reasons. Firstly, traditional crowdfunding platforms face inherent limitations, such as intermediaries, lack of transparency, and susceptibility to fraud. By leveraging blockchain, we can overcome these challenges and create a more efficient, transparent, and secure crowdfunding ecosystem. The implementation of a blockchain-based crowdfunding platform eliminates the need for intermediaries, allowing for direct peer-to-peer interactions between project creators and backers. This decentralization reduces costs, removes unnecessary gatekeepers, and ensures that funds reach the intended recipients without delays or additional fees. Transparency is a critical aspect of any crowdfunding initiative. Blockchain's distributed ledger technology provides a transparent record of transactions, enabling backers to track the flow of funds and verify the authenticity and progress of projects. This transparency builds trust between project creators and backers, increasing confidence and participation in crowdfunding campaigns.

Fraud is a significant concern in traditional crowdfunding. Blockchain's immutable and tamper-resistant nature mitigates the risk of fraudulent activities, ensuring that contributions are securely stored and transactions are verifiable. Smart contracts, programmable code executed on the blockchain, can automate the distribution of funds based on predefined criteria, minimizing the potential for misappropriation or misuse. Furthermore, implementing blockchain technology promotes financial inclusion by accepting various cryptocurrencies as contributions. This approach eliminates barriers associated with traditional banking systems and enables individuals without access to banking services or in underserved regions to participate in crowdfunding campaigns. It opens up opportunities for a global community of backers, empowering creators from all backgrounds to access the necessary resources for their projects.

Muneeza , et al.[1] begins by providing an overview of crowdfunding, which is a type of alternative finance that allows individuals or groups to raise money from a large number of people, typically through online platforms. Crowdfunding has become increasingly popular in recent years, as it offers a number of advantages over traditional forms of finance, such as lower costs, greater flexibility, and increased transparency. The journal then discusses the potential benefits of using blockchain technology in crowdfunding. Blockchain is a distributed ledger technology that can be used to record transactions in a secure and transparent manner. This could make crowdfunding more efficient and secure, and could also help to reduce fraud and other risks. Additionally, blockchain could make crowdfunding more accessible to people in developing countries, who may not have access to traditional financial services. The journal concludes by discussing the challenges and limitations of using blockchain technology in crowdfunding. One challenge is that blockchain is still a relatively new technology, and there is a lack of understanding about how it works. Additionally, blockchain can be expensive to implement, and it may not be suitable for all types of crowdfunding projects. Overall, the journal provides a positive view of the potential of blockchain technology to improve crowdfunding. However, it is important to note that blockchain is still a new technology, and there are a number of challenges that need to be addressed before it can be widely adopted.

Pandey , et al.[2] discusses crowdfunding is a new way to raise money for projects. It allows people to donate small amounts of money to projects they care about. Crowdfunding has become increasingly popular in recent years. In 2018, the global crowdfunding market was worth $34.4 billion. However, crowdfunding is also a target for fraud. In 2018, there were over 1 billion in crowdfunding fraud cases. The authors believe that their blockchain-based crowdfunding platform would be a more secure and transparent way to raise money for projects. They also believe that it would help to reduce crowdfunding fraud. This paper provides a comprehensive overview of the research on blockchain and crowdfunding. It identifies the key challenges and opportunities of using blockchain for crowdfunding. It also discusses the potential benefits of using blockchain for crowdfunding, such as increased security, transparency, and efficiency. Overall, the research on blockchain and crowdfunding is still in its early stages. However, there is a growing body of research that suggests that blockchain has the potential to revolutionize crowdfunding. Blockchain can help to make crowdfunding more secure, transparent, and efficient. It can also help to reduce crowdfunding fraud. As the research on blockchain and

crowdfunding continues, it is likely that we will see even more innovative and secure ways to use blockchain for crowdfunding.

Yadav , et al.[3] discusses Blockchain and crowdfunding and are two emerging technologies that have the potential to revolutionize the way we raise and invest money. Crowdfunding is a way to raise money for projects by getting small amounts of money from a large number of people. Blockchain is a distributed ledger technology that can be used to store data in a secure and transparent way. The combination of blockchain and crowdfunding could create a more secure, transparent, and efficient way to raise money for projects. Blockchain could be used to store data about crowdfunding projects, such as the project's goals, timeline, and budget. This information would be available to everyone, which would help to reduce fraud and make it easier for people to make informed investment decisions. Blockchain could also be used to automate crowdfunding transactions. This would make it easier for people to invest in projects and would help to reduce the costs associated with crowdfunding. Overall, the combination of blockchain and crowdfunding has the potential to create a more efficient and secure way to raise money for projects. This could lead to a more vibrant and innovative economy.

Gururaj, et al.[4] defines crowdfunding as a way to raise money for projects by getting small amounts of money from a large number of people. Blockchain is a distributed ledger technology that can be used to store data in a secure and transparent way. The authors of the paper propose a decentralised application for crowdfunding using blockchain technology. The application would use smart contracts to automate crowdfunding transactions and ensure that funds are only released to legitimate projects. The application would also use blockchain to store data about crowdfunding projects, such as the project's goals, timeline, and budget. This information would be available to everyone, which would help to reduce fraud and make it easier for people to make informed investment decisions. The authors of the paper believe that their decentralised application for crowdfunding using blockchain technology would be a more secure, transparent, and efficient way to raise money for projects. They also believe that their application would be more accessible to people in developing countries. The paper is well-written and the authors make a convincing case for the use of blockchain in crowdfunding. However, the paper does not provide any empirical evidence to support its claims. It would be interesting to see a study that compares the performance of the proposed decentralised

application to traditional crowdfunding platforms. Overall, the paper is a valuable contribution to the literature on blockchain and crowdfunding. It provides a clear and concise overview of the technology and its potential applications in crowdfunding. The authors make a strong case for the use of blockchain in crowdfunding and their proposed application is a promising development.

Bogusz, et al.[5] begins by providing an overview of each of the technologies. Crowdfunding is a method of raising money from a large number of people, typically through online platforms. Blockchain technologies are distributed ledger systems that can be used to record transactions in a secure and transparent manner. Cryptocurrencies are digital or virtual tokens that use cryptography to secure their transactions and to control the creation of new units. ICOs are a type of crowdfunding that allows companies to raise capital by selling tokens that represent ownership in the company. The authors then discuss the potential benefits of these technologies for entrepreneurial finance. Crowdfunding can help entrepreneurs to raise capital from a wider range of investors, including individuals who may not have access to traditional sources of funding. Blockchain technologies can help to reduce the costs of raising capital and can make the process more transparent. Cryptocurrencies can provide a more efficient and secure way for investors to participate in early-stage ventures. ICOs can provide a way for companies to raise capital without having to go through traditional venture capital firms. The authors conclude by discussing the challenges and opportunities associated with the use of these technologies for entrepreneurial finance. One challenge is that these technologies are still in their early stages of development and there is a lack of regulation. Another challenge is that these technologies can be complex and difficult to understand. However, the authors argue that the potential benefits of these technologies outweigh the challenges. They believe that these technologies have the potential to revolutionize entrepreneurial finance and to make it easier for entrepreneurs to raise capital and for investors to participate in early-stage ventures.

Here are some of the key points from the paper:

- Crowdfunding, blockchain technologies, cryptocurrencies, and ICOs are all emerging technologies that have the potential to revolutionize entrepreneurial finance.
- These technologies can help entrepreneurs to raise capital from a wider range of investors, reduce the costs of raising capital, and make the process more transparent.

- These technologies can also provide a more efficient and secure way for investors to participate in early-stage ventures.

Basu, et al.[6] propose a system for crowdfunding NFTs using blockchain technology. The system would allow artists to create and sell NFTs to a wider audience. The system would also allow investors to buy and sell NFTs. The system would be more secure and transparent than traditional crowdfunding platforms. The paper is well-written and the authors make a convincing case for the use of blockchain in crowdfunding NFTs. However, the paper does not provide any empirical evidence to support its claims. It would be interesting to see a study that compares the performance of the proposed system to traditional crowdfunding platforms. Overall, the paper is a valuable contribution to the literature on blockchain and crowdfunding. It provides a clear and concise overview of the technology and its potential applications in crowdfunding NFTs. The authors make a strong case for the use of blockchain in crowdfunding NFTs and their proposed system is a promising development.

Khatter, et al.[7] propose a system for secure and transparent crowdfunding using blockchain technology. The system would use smart contracts to automate crowdfunding transactions and ensure that funds are only released to legitimate projects. The system would also use blockchain to store data about crowdfunding projects, such as the project's goals, timeline, and budget. This information would be available to everyone, which would help to reduce fraud and make it easier for people to make informed investment decisions. The authors of the paper believe that their system for secure and transparent crowdfunding using blockchain technology would be a more secure, transparent, and efficient way to raise money for projects. They also believe that their system would be more accessible to people in developing countries. The paper is well-written and the authors make a convincing case for the use of blockchain in crowdfunding. However, the paper does not provide any empirical evidence to support its claims. It would be interesting to see a study that compares the performance of the proposed system to traditional crowdfunding platforms. Overall, the paper is a valuable contribution to the literature on blockchain and crowdfunding. It provides a clear and concise overview of the technology and its potential applications in crowdfunding. The authors make a strong case for the use of blockchain in crowdfunding and their proposed system is a promising development.

Berawi, et al.[8] propose a blockchain-based crowdfunding model for property investment. The model would use smart contracts to automate crowdfunding transactions and ensure that funds are only released to legitimate projects. The model would also use blockchain to store data about property investment projects, such as the project's goals, timeline, and budget. This information would be available to everyone, which would help to reduce fraud and make it easier for people to make informed investment decisions. The authors of the paper believe that their blockchain-based crowdfunding model for property investment would be a more secure, transparent, and efficient way to raise money for property investment projects. They also believe that their model would be more accessible to people in developing countries. The paper is well-written and the authors make a convincing case for the use of blockchain in property investment crowdfunding. However, the paper does not provide any empirical evidence to support its claims. It would be interesting to see a study that compares the performance of the proposed model to traditional property investment crowdfunding platforms. Overall, the paper is a valuable contribution to the literature on blockchain and property investment crowdfunding. It provides a clear and concise overview of the technology and its potential applications in property investment crowdfunding. The authors make a strong case for the use of blockchain in property investment crowdfunding and their proposed model is a promising development.

Bracamonte, et al[9] explore the influence of guidelines on crowdfunding projects in the Ethereum blockchain platform. They analyze data from 1,000 crowdfunding projects on the Ethereum platform and find that projects that follow the guidelines are more likely to be successful.

The authors identify three key guidelines that are associated with project success:

- Clear and concise project description: The project description should be clear and concise, and it should provide potential investors with all the information they need to make an informed decision.
- Transparent project management: The project manager should be transparent about the project's progress, and they should keep potential investors updated on any changes or delays.
- Active community engagement: The project manager should engage with the community and answer any questions that potential investors may have.

Chapter 3

Methodology

3.1 System Requirements

3.1.1 Hardware Requirements

- Processor : 11th Gen Intel® Core™ i5-11300H @ 3.10GHz, 3110 Mhz, 4 Core(s), 8 Logical Processor(s)
- RAM : 4 GB or Above
- Hard Disk : 512 GB or Above
- Internet : 4Mbps or above (Wired or Wireless)
- Display : 15.5” Color Monitor
- Screen Resolution : 1920 x 1080 x 60 Hz
- Color Palette : True Color (32 bit)
- Keyboard : PC/AT enhanced type
- Mouse : Zenronics Zeb-Transformer-M Optical/HID

3.1.2 Software Requirements

- Operating System : Any OS capable of running a browser (Mac, Windows,Linux)
- Front-end : React Js (HTML, JS), Tailwind CSS

- Back-end : Blockchain
- Libraries : Web3.js, ether.js,
- Language : JavaScript, Solidity
- Tools Required : Metamask (Browser Extension), npm ,Hardhat-Ethereum IDE
- IDEs : VS-Code
- Hosting : Github (code) Netlify (website hosting)

3.2 System Design

3.2.1 System Architecture

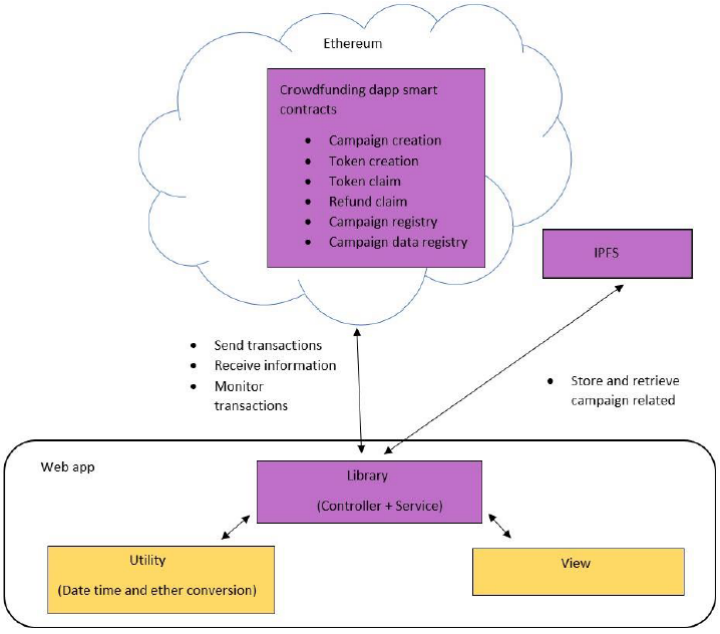

Figure 3.1: A basic crowdfunding dapp architecture

The basic architecture of the crowdfunding dapp is depicted in the Figure 1 a basic crowdfunding dapp architecture diagram, concentrating on high-level components. All interactions between a campaign creator (a person arriving in the platform to raise funds) and a campaign investor (a person arriving in the platform to invest ether) are mediated by the smart contracts written for crowdfunding dapp deployed in Ethereum blockchain platform. For example if an investor, wants to invest certain amount of Ether in a particular campaign that interests him,

a transaction is initiated and sent to Ethereum network with additional Ether to pay for the mining fees.

Creating a campaign has other meta-data associated with it such as the campaign description, detailed plan and concept about the campaign, images and videos to describe the concept, documents such as white paper and creator or team name. All these meta-data are stored in IPFS and fetched by the library component when the application loads to supply to the view component. The view component is responsible for building the HTML view displayed to the user.

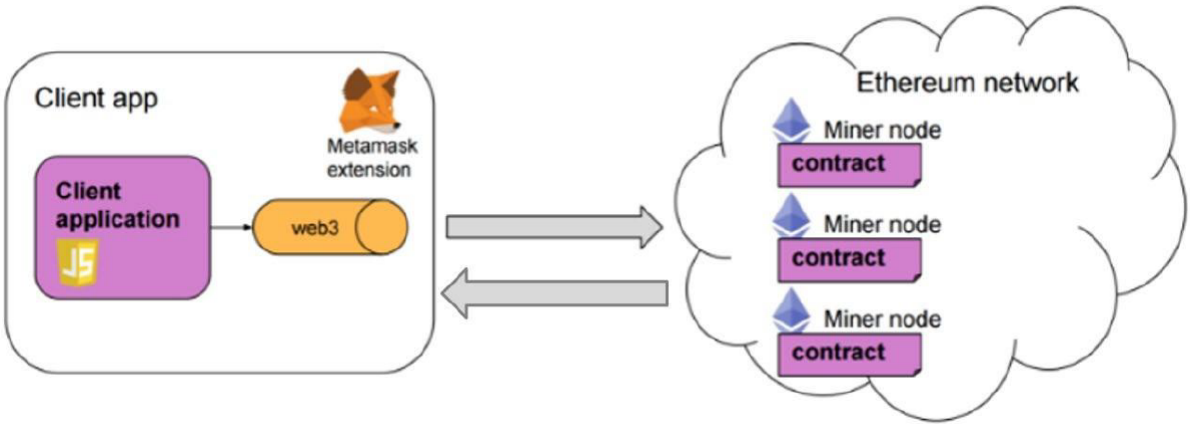

Figure 3.2: Library component using Metamask to communicate with Ethereum nodes

As shown in Figure 2 library component using Metamask to communicate with Ethereum nodes library component in the dapp uses web3 (Ethereum JavaScript API) to trigger transactions depending on the user interaction with the view component. Metamask acts as lightweight client that provides async functionality to library component for making calls to Ethereum network using API's provided by web3. Metamask also tracks the transaction processing and updates the library component about the status of mining of that transaction. A transaction needs to be signed before it can be sent to Ethereum blockchain. Metamask handles this by using the API's provided in web3 library. Once the transaction is signed Metamask will make a RPC call to send the transaction to Ethereum network where the smart contracts are deployed. Metamask polls Ethereum network to check the status of the transaction and update the Library component using a callback function present in library component. The other network calls handled by library component is to the IPFS to store and retrieve campaign metadata. IPFS provides standard set of API's to interact with its file system. Library component makes web service calls to IPFS to store the metadata and receive a link to the

stored data so in future it can just make a web call to retrieve that data. The library component builds a data object with details received from IPFS and Ethereum network; it converts the data to a format suitable for view component to render in the browser. To perform this data conversion library uses the utility component that has many stateless helper functions. These helper functions take input in certain format such as big number and convert it to UI renderable strings. The view component becomes lightweight by moving all the data formatting logic to utility component. Library component mainly acts as the controller and service module as per the MVC architecture. View component is comprised mostly of HTML templates rendered in the browser. It receives all the data needed from library component, then parses it to get the exact values to be replaced in the dynamic HTML templates. It also takes care of the routing of pages and style sheets required to beautify the web page. Event handlers are in place to respond for any user actions such as button clicks and drop-down changes. Library component is tightly coupled with view component so any changes in the UI can be bubbled to library component for communicating it to blockchain.

3.2.2 User Interface

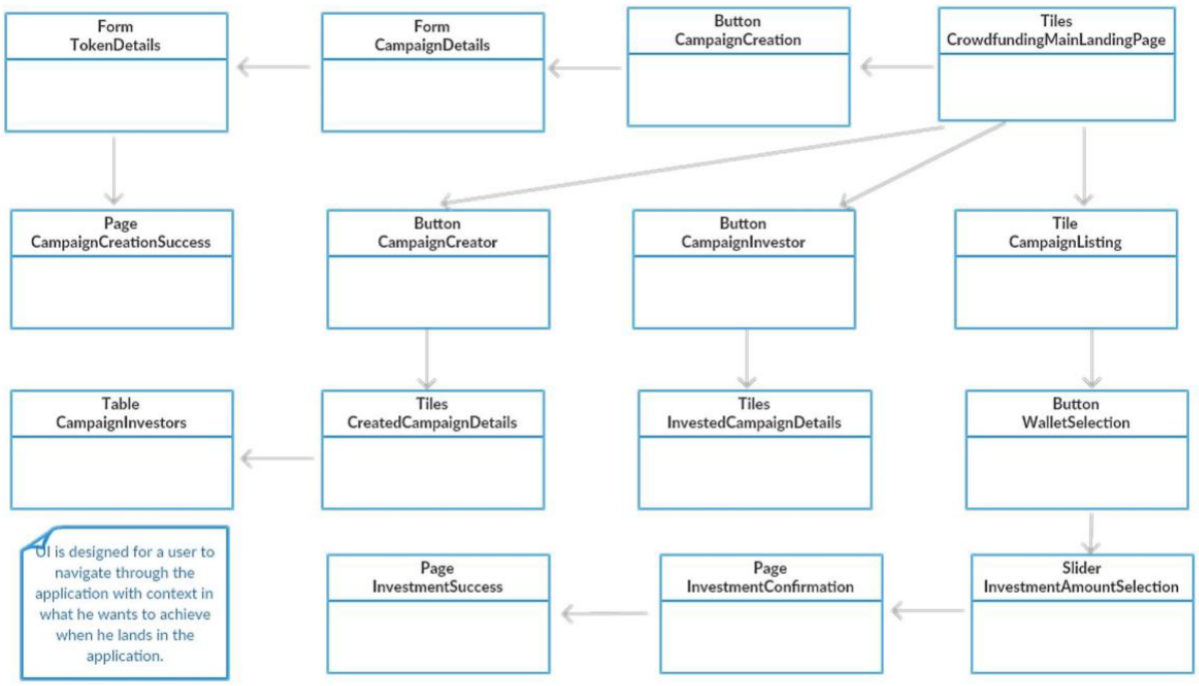

Figure 3.3: crowdfunding app user interface design

The user interface is designed in keeping mind the ease for a campaign creator to create a new campaign and an investor to invest in that campaign. The landing page of the application is the root of the user interface as seen in the Figure 3 crowdfunding app user interface design. It is made up of campaign tiles briefing about each campaign that is listed on the platform. User can sort the campaigns based on category each campaign belongs to. Also, user is provided with a search option to shortlist campaigns based on various search criteria. When a user clicks on a campaign tile the campaign details page containing videos, images, documents and detailed descriptions of the campaign is displayed. A button to contribute to the campaign is displayed using which the user can select the wallet details from MetaMask. Then the user is presented with a slider to adjust the amount of ether he wants to contribute to the campaign. After he fills in the amount, he wants to contribute to a campaign then he navigates to confirmation page to see his final contribution details. A user transaction is triggered in MetaMask which will send the transaction to Ethereum network. Once a confirmation about the campaign is received the confirmation page redirects to a transaction investment success page to display a success message. A campaign investor to track his investments can navigate to the investor details page from the landing page using the campaign investor button. He can find all the details about campaigns that he has invested in and if he has to claim the tokens or if he needs to reclaim his investment in case a campaign fails. For creating a campaign, the user can navigate to register campaign page from the landing page using the register a campaign button. After filling in the details about his campaign in the campaign creation form then he can navigate to the token details page to fill campaign specific token details. After entering all the details, he can register the campaign by clicking on register button which takes him to the progress page to track the campaign creation progress. Once the campaign is created, to track the details about investors and campaign progress a campaign creator can use the campaign creator button in the landing page. It consists of tiles representing campaigns created by a logged in creator. When a tile is clicked it navigates to the investor details page showing all the investors of that campaign with their respective investment details and a button to claim the invested amount if campaign is successful.

3.2.3 Ethereum Smart Contract Design

The smart contracts written for crowdfunding dapp tie the whole platform together and make it work by controlling the flow of business logic. Even though they do not contain any of the bulk

data, the business model they define and transactions that change the business logic flow are how this system of crowdfunding can work in a peer-to-peer manner, without central control.

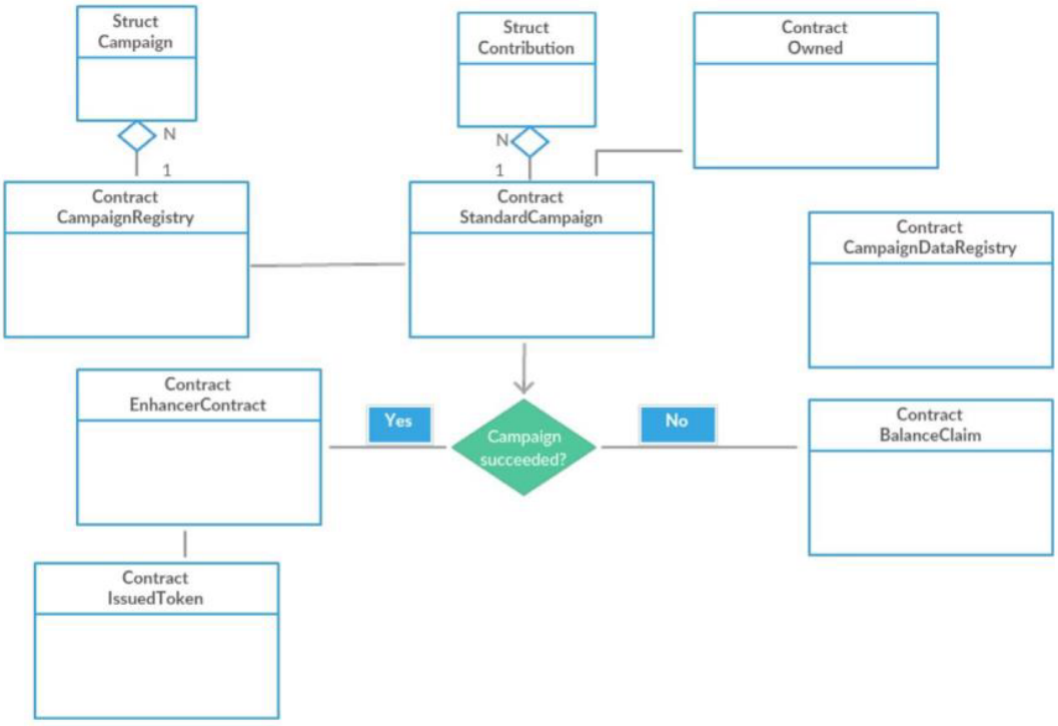

Figure 3.4: Crowdfunding dapp smart contract design

The layout of crowdfunding dapp smart contracts is shown in Figure 4 crowdfunding dapp smart contract design. The root of the platform is CampaignRegistry contract, which is created only once and contains all the campaigns reference that are created in the platform. When a new campaign contract is created its contract address and contract creation timestamp is stored in the Campaign registry contract. The crowdfunding dapp platform fetches all the contract references, stored in a struct datatype, from Ethereum network when the app is loaded and builds the landing page with campaign tiles. The CampaignDataRegistry contract contains information about IPFS hash, where a contracts metadata is present, so that when a campaign details page has to be built the crowdfunding dapp can call this contract and use the reference returned to fetch a campaign's images, videos and other details. It consists of a map that takes in a campaign's address and returns the corresponding IPFS hash. IssuedToken contract contains details pertaining to a campaign specific token that is issued to the investor. It is associated with an enhancer contract that triggers the transfer of tokens to a particular investor if the campaign succeeds. ModelEnhancer contract as the name suggests is an enhancer contract that contains details about each contribution to the campaign and calculates the number of campaign

specific tokens to be issued in case the campaign succeeds. StandardCampaign contract is the most important contract that handles the business logic flow of a campaign. It contains the contributor contribution details and stage information of a campaign such as success, operational and failure details. Every campaign when created will have its own standard campaign contract that guides the execution of the campaign. Depending upon the result of funding standard campaign contract triggers ModelEnhancer contract to disperse camping specific tokens or creates a balance claim contract to transfer ether that has to be refunded to the investors. Incase of a campaign failure standard campaign contract triggers multiple claim balance contracts for each investor to get his investment back. Standard campaign uses a struct data type called contribution to record all the contributions made to a campaign. It refers to this datatype in case of a camping failure to create the balance claim contracts.

3.3 System Implementation

Crowdfunding dapp is divided into various activates a user can perform in the platform. Anyone who lands on the platform can browse through all the campaigns listed and explore more about each campaign. The application extensively uses Javascript libraries to build the user interface, handle user inputs and communicate with Ethereum network. The landing page uses CampaignRegistry contract to get all the campaigns registered on the platform. Once it gets the campaigns then it accrues campaign related data from IPFS and builds the user interface of the landing page.

3.3.1 Campaign Listing

A campaign struct datatype containing three data fields i.e. address of the campaign, abi(application binary interface gives a standard interface for calling particular function in a smart contract) and time the campaign is created are stored in the CampaignRegistry contract. It consists of an array named campaigns of type campaign which is called by the dapp to get all the campaigns. It also consists of other auxiliary functions that are used while building individual campaign pages. To get the total number of campaigns created in the platform the web application calls the numberOfCampaigns() function. If a contract's function is declared public then it can be called by any Ethereum network user or by any contract with the contract's address. The registeredAtTimeStamp() function takes a campaignID which

is a number as a input and gives the time at which a campaign is created. During the view construction these pieces of information is assembled by the library and fed to the view component which populates the HTML template rendered on the browser. The view component calls getCampaigns() function by passing an array of campaign ids got by calling the numCampaigns() function in campaign registry contract. getCampaigns() iterates over those campaign ids ,which are simple array indexes, to get the corresponding campaign object having the campaign contract address fetching campaign details client code. Using the campaign address retrieved from the campaign registry contract the library component creates an instance of Standard campaign contract to get the campaign specific details such as the expiry, funding goal, funding cap, owner and amount raised. Once the getCampaings() accrues all the data then it forwards a assembled campaign object to view component.

3.3.2 Campaign Details

The view component calls the library component to get all the details to build the campaign details page. The library component creates an instance of the standard campaign contract and fetches campaign specific details such as name, funding goal, expiry, owner, beneficiary and funding cap. After getting these campaign specific details then it uses the campaign data registry contract to get the IPFS hash using which it can fetch any metadata related to the campaign.

```
//get the owner of campaign
contracts.Owned.factory.at(campaignAddress).owner(function (oError, oAddress) {
  //detect any error
  if (oError) {
    return callback(oError, null);
  }
  //set the owner address
  returnedCampaignObject.owner = oAddress;
  //get amount invested in campaign
  web3.eth.getBalance(campaignAddress, function (bError, bResult) {
    //detect any error
    if (bError) {
      return callback(bError, null);
    }
    //set amount invested in campaign
    returnedCampaignObject.balance = bResult;
```

Figure 3.5: Get the campaign owner and amount raised

The address of the owner and the amount raised by a campaign can be got by accessing the campaign's contract in the Ethereum network as shown in Figure 15 get the campaign owner and amount raised. Similarly, other details such as:

- Campaign token address
- Campaign enhancer address
- Contributions to a campaign
- Campaign Expiry

3.3.3 Campaign Investment

Ether can be invested in a campaign by an investor using "Back this project" button. This button is associated with an event handler which navigates the page to wallet selection page where a button is provided for the investor to confirm his usage of MetaMask. After the confirmation of his usage of MetaMask then the page navigates to contribution form to allow the investor to specify his contribution amount. User MetaMask account balance is checked to see if he has sufficient funds to invest in that campaign. Also, the token price is fetched to calculate the amount of tokens to be returned to the investor in case the campaign is successful.

```
//get the enhancer contract instance
var enhancer = contracts.Model1Enhancer.factory.at(campaignData.enhancer);
var pollTokenPrice = () => {
  enhancer.price((error, result) => {
    if (!error && result) {
      //ether value
      var etherValue = new BigNumber(web3.fromWei(result, 'ether'));
      //append the ether value
      el('#contributeTokenPrice').innerHTML = '';
      el('#contributeTokenPrice').appendChild(yo`<span>${etherValue.toString(10)}</span>`);
      el('#campaignContributeAmount').value = etherValue.toString(10);
      //Set the token price in environment.js for global access
      setTokenPrice(etherValue);
    }
  });
};
```

Figure 3.6: Client code to fetch the token price

The Figure 6 client code to fetch the token price shows how the client code creates an instance of enhancer contract and calls the price function to get the price of the token. Once the price is fetched then the slider to toggle the investment amount is configured. The investor can use the slider to increase or decrease the amount he wants to invest using this slider. Once the investor decides the amount, he wants to invest then he clicks on review contribution button to get a summary about his contribution. Campaign contribution shows a sample review of contribution to a campaign. Once the investor verifies all the contribution details then he can make the contribution using Make Contribution button which triggers MetaMask to send a campaign contribution transaction. Once the transaction is successful then the page is navigated to contribution receipt page confirming successful contribution

3.3.4 Campaign Creation

For a user to create a campaign he can use the "Register Campaign" button. There is a form that must be filled up by the campaign creator. This form has all the fields that are used to build the campaign details page and the campaign tile present in the landing page. Below listed are the fields present in the campaign details form:

- Title of the campaign
- Target amount (ETH)
- Campaign Expire date
- Banner image for the Campaign
- Description about the campaign

The campaign creator needs to fill in all the fields and upload the appropriate image. Then user can proceed to the next form where he is given a set of fields that pertain to campaign contract, model enhancer contract and token contract creation. Once all the details are filled the campaign creator clicks on "Submit Project" button that triggers the handleCampaignRegister event handler that will initiate transactions to the Ethereum network.

```
//Campaign creation constructor
function StandardCampaign(string campaignName,
  uint256 campaignExpiry,
  uint256 campaignFundingGoal,
  uint256 campaignFundingCap,
  address campaignBeneficiary,
  address campaignOwner,
  address campaignEnhancer) public {

  name = campaignName;
  expiry = campaignExpiry;
  fundingGoal = campaignFundingGoal;
  fundingCap = campaignFundingCap;
  beneficiary = campaignBeneficiary;
  owner = campaignOwner;
  created = block.number;
  enhancer = Enhancer(campaignEnhancer);
}
```

Figure 3.7: Standard Campaign constructor contract code

HandleCampaignRegister uses the new function to create a new campaign. As seen in the Figure 7 standard campaign constructor contract code there are seven parameters needed to create a new campaign. HandleCampaignRegister fetches all the required values from both the forms and constructs a request that is sent to the Ethereum network using MetaMask. After the campaign contract is mined the campaign specific token is created according to the parameters provided by the campaign creator. Issued token consists of the token name and symbol token constructor contract code used by the wallets to represent the token in them. Token issuance is moderated by an enhancer contract that keeps track of contributors and the tokens that have to be issued to them depending upon the completion of funding. Enhancer contract has claim method that is used to claim the token upon successful funding of the campaign

3.3.5 Campaign Dashboard

```
async function displayContribution(currentCampaignContributions) {
    //iterate through all the contrbutions
    for(var i = 0; i < currentCampaignContributions.length; i++ ) {
        var contributor = currentCampaignContributions[i];
        //get the trasnasction time
        var timestampPromise =  new Promise ((resolve, reject) => {
            web3.eth.getBlock(contributor[2], (err, blockDetails) => {
                resolve(new Date(blockDetails.timestamp * 1000));
            });
        });

        var timestampOfTransaction = await timestampPromise;
        //pass the details to the template
        el('#contibutors-list').appendChild(campaignContributorList({
            contributor: currentCampaignContributions[i],
            count: i+1,
            timestamp: timestampOfTransaction,
        }));
    }
}
```

Figure 3.8: Client code to populate a contribution object and pass it to the template

For a campaign creator to track all his campaigns and get details about contributions made to a campaign he can use the campaign creator dashboard. The view component calls the getCampaign() method of library component which in turn calls the contributions array present in the campaign contract to fetch the contributions. Library component returns the contribution array to the event handler which parses each contribution and fetches the block creation time in which the contribution transaction is present. Once it gets all the details needed by the page template the library component passes those details over to template code to populate a contribution object and pass it to the template which is rendered to the user.

3.4 Technology Used

3.4.1 Blockchain

Blockchain is a distributed ledger system that allows for secure, decentralized transactions. The technology was first developed as the foundation of the cryptocurrency, Bitcoin. However, its potential uses go far beyond the world of finance. Blockchain works by creating a

permanent, unalterable record of transactions. It does this by using a network of computers to validate and confirm each transaction. The main features of blockchain technology include decentralization, immutability, transparency, and security. Decentralization means that there is no central authority controlling the system. Instead, the blockchain is maintained by a network of computers, making it more resistant to attacks and tampering. Immutability refers to the fact that once a transaction is recorded on the blockchain, it cannot be altered or deleted. Transparency means that every transaction on the blockchain is visible to all participants. Finally, security is achieved through the use of cryptography, which ensures that transactions are secure and private Crowdfunding platforms are online platforms that allow entrepreneurs and businesses to raise funds from a large number of people. Blockchain technology can be implemented in crowdfunding platforms in a number of ways. For example, it can be used to create smart contracts that automate the process of crowdfunding. These smart contracts can be programmed to automatically release funds when certain conditions are met, such as when a certain amount of money is raised. Blockchain can also be used to increase transparency in the crowdfunding process. By recording all transactions on a public ledger, blockchain technology can ensure that all investors have access to the same information about the project they are funding. This can help to build trust between investors and entrepreneurs. There are many benefits to using blockchain technology in crowdfunding platforms. One of the main benefits is increased transparency. By recording all transactions on a public ledger, blockchain technology can ensure that investors have access to the same information about the project they are funding. This can help to build trust between investors and entrepreneurs. Another benefit of using blockchain technology in crowdfunding is increased security. Blockchain technology uses cryptography to ensure that transactions are secure and private. This can help to prevent fraud and protect investors' funds. Blockchain technology can help to automate the crowdfunding process. By creating smart contracts that automate the process of crowdfunding, entrepreneurs can save time and reduce the administrative burden of raising funds.

3.4.2 Smart Contracts

Smart contracts are self-executing digital contracts that are programmed to automatically execute when certain conditions are met. They are a key feature of blockchain technology and have the potential to revolutionize the way we think about contracts and transactions. The main features of smart contracts in blockchain include automation, transparency, and security.

Automation means that the contract is automatically executed when certain conditions are met, which can help to reduce the need for intermediaries and increase the efficiency of transactions. Transparency means that all parties to the contract have access to the same information, which can help to reduce disputes and build trust. Finally, security is achieved through the use of cryptography, which ensures that the contract is secure and tamper-proof. Smart contracts can be implemented in crowdfunding platforms in a number of ways. For example, they can be used to automate the process of crowdfunding, by setting up a contract that automatically releases funds when certain conditions are met, such as when a certain amount of money is raised. If the campaign deadline is reached and the target amount is not met, the smart contract can automatically refund the raised amount to the respective investors. Additionally, if the campaign creator decides to delete their campaign, the smart contract can automatically release funds back to the investors. Overall, smart contracts are a powerful tool for implementing blockchain technology in crowdfunding platforms. They offer a number of benefits, including increased efficiency, transparency, and security, and have the potential to revolutionize the way we think about contracts and transactions.

3.4.3 Hardhat

Hardhat is a development environment for building and testing smart contracts on the Ethereum blockchain. It is an open-source tool that is widely used by developers to create and deploy decentralized applications (dApps). Hardhat offers a range of features that make it an ideal tool for building blockchain applications. In the context of crowdfunding platforms, Hardhat is used to build and test the smart contracts that automate the fundraising process. It enables developers to quickly and easily write and test code for their contracts, which can help to reduce development time and improve the quality of the final product. Overall, Hardhat is a powerful tool for building decentralized applications on the Ethereum blockchain, and its features make it an ideal choice for use in crowdfunding platforms.

3.4.4 Web.js

Web3.js is a JavaScript library that provides a way to interact with Ethereum and other blockchain networks. It is an open-source project that enables developers to build decentralized applications (dApps) that run on the blockchain. Web3.js offers a range of features that make

it an ideal tool for building blockchain applications. The main features of Web3.js include the ability to interact with smart contracts, read and write data to the blockchain, and send and receive cryptocurrency. It also includes a number of built-in functions and utilities that make it easy to develop dApps, including support for Ethereum accounts, transactions, and events. Additionally, Web3.js provides a user-friendly API that simplifies the process of interacting with the blockchain. Web3.js is used to interact with smart contracts that automate the fundraising process. It enable to build user interfaces that allow users to easily contribute to campaigns, track their contributions, and view campaign progress. Additionally, Web3.js can be used to securely manage user accounts and interact with the blockchain to execute transactions.

3.4.5 Metamask

MetaMask is a chrome browser plugin that acts as a bridge between your browser and Ethereum blockchain by providing a secure identity vault, a user interface to manage multiple Ethereum wallets and sign blockchain transactions. It is one of the best ways to send transactions to Ethereum blockchain because it keeps a track of transaction execution and returns if any error occurs during mining or execution. It supports any ERC20 type token to be added to your wallet and trigger transaction on those ERC20 tokens. It is an Ethereum community open source project having more than million active users; hence, it is the most popular plugin to interact with blockchain.

3.4.6 Testnet

Test network (Testnet) is a copy of Ethereum blockchain identical in every way to main network except the fact that their Ether is worthless. There are three types of testnets public, private and GanacheCLI. As names suggest, public testnet are available to everyone and connected to the internet, private testnet are similar to one's own blockchain and GanacheCLI is a simulation of Ethereum network on a single computer. For this project, we are using a public testnet called Goerli (for deployed version) and private testnet localhost 8545.

3.4.7 IPFS

IPFS stands for interplanetary file system. It is a protocol and network used to store and share hypermedia in a distributed file system. It is an open source project maintained by a huge community of developers. The contents in IPFS are accessed in two ways, via FUSE (in case of Linux) and over HTTP even though IPFS wants to replace HTTP. IPFS can be seen as a BitTorrent swarm, exchanging objects within a single Git repository

3.5 Software and Language Specifications

3.5.1 React JS

React.js is an open-source JavaScript library that is widely used for building user interfaces for web applications. Developed by Facebook, React.js enables developers to build complex and dynamic user interfaces that can be updated in real-time without requiring a full page refresh. React.js is based on a component-based architecture, which means that the user interface is broken down into smaller reusable components. This makes it easier to manage and scale complex applications, as each component can be developed and tested independently. React.js is also known for its high performance, as it uses a virtual DOM (Document Object Model) to efficiently update the user interface without requiring a full page reload. Overall, React.js is a powerful and versatile tool for building modern web applications, and its popularity has led to a large and active developer community that continues to innovate and improve the library.

React Features

- Component-Based Architecture: React.js is based on a component-based architecture, which makes it easy to create reusable UI components that can be used throughout the application. This is particularly useful in the context of blockchain applications, as it allows developers to break down complex UI elements into smaller, more manageable components.
- Virtual DOM: React uses a virtual representation of the actual DOM, known as the Virtual DOM. The Virtual DOM allows React to efficiently update and render only the necessary parts of the UI, leading to better performance.

- Unidirectional data flow: React follows a unidirectional data flow, also known as one-way data binding. Data flows from parent components to child components, preventing unexpected side effects and making the application easier to understand and debug.

- JSX: React uses JSX (JavaScript XML), which is an extension to JavaScript that allows you to write HTML-like syntax within JavaScript code. JSX simplifies the process of creating and manipulating UI components.

- Component lifecycle methods: React components have lifecycle methods that allow you to hook into different stages of a component's life, such as when it is being created, updated, or destroyed. These methods provide hooks for performing actions at specific points in the component's lifecycle.

- React Hooks: Introduced in React 16.8, hooks are functions that allow you to use state and other React features in functional components. Hooks provide a way to reuse stateful logic without writing a class component, making code more concise and easier to understand.

React Library

React libraries are collections of pre-built and reusable code components that extend the functionality of React.js, a popular JavaScript library for building user interfaces for web applications.

- web3.js: A library that provides a way to interact with the Ethereum blockchain, including the ability to read and write data to smart contracts.

- Drizzle: A library that provides a simple way to connect React.js applications to the Ethereum blockchain, including the ability to listen for events and interact with smart contracts.

- React Router: A library that provides routing capabilities for React.js applications, allowing for easy navigation between different pages and components

- IPFS: A decentralized storage system that can be used to store and distribute files on the blockchain, including content related to crowdfunding campaigns.

3.5.2 Solidity

Solidity is a programming language used for writing smart contracts, which are self-executing contracts with the terms of the agreement directly written into code. Solidity is a statically-typed language, meaning that variables must be explicitly declared with their data types. It also supports object-oriented programming (OOP) concepts like inheritance and polymorphism, which can make it easier to write complex smart contracts. Solidity was developed specifically for the Ethereum platform, and includes features like gas (the unit of measurement for transaction fees on the Ethereum network) and the ability to interact with other contracts on the blockchain. Solidity contracts are compiled into bytecode, which can then be deployed to the Ethereum network and executed by Ethereum nodes.

Solidity Features

- Turing completeness: Solidity is a Turing-complete language, which means that it can be used to write any program that can be written in any other programming language. This allows for the creation of complex smart contracts that can perform a wide range of functions on the blockchain.
- Security: Solidity is designed with security in mind, and includes features like built-in checks for integer overflow and underflow, as well as mechanisms for preventing reentrancy attacks and other common security vulnerabilities.
- Gas optimization: Solidity includes features that allow developers to optimize their smart contracts for gas usage, which is a key consideration in the context of blockchain development. By minimizing gas usage, developers can help to reduce the cost of executing their smart contracts on the blockchain.
- Interoperability: Solidity is designed to be interoperable with other smart contracts on the blockchain, allowing for the creation of complex decentralized applications that can interact with each other in a seamless and efficient manner.
- Ethereum-specific functionality: Solidity is designed specifically for use on the Ethereum blockchain, and includes features like the ability to create and manage ERC-20 tokens, interact with other smart contracts on the blockchain, and store data in a decentralized manner.

Solidity Library

Solidity libraries are smart contracts themselves, but unlike regular contracts, they don't have a storage of their own. Instead, they are deployed as a separate bytecode and can be linked to other contracts at compile time.

- SafeMath: SafeMath is a library that provides arithmetic functions with overflow and underflow checks to prevent errors in integer calculations. This library is especially important when dealing with token balances and calculating funding amounts in a crowdfunding platform.
- Crowdsale: The Crowdsale library is a simple and flexible way to create a token sale contract. It allows developers to set the token price, the funding goal, the minimum and maximum contributions, and other parameters.
- PaymentSplitter: The PaymentSplitter library is a contract that allows for the automatic distribution of payments to multiple parties. This library can be used to split the funds raised in a crowdfunding campaign among the project team members, investors, and other stakeholders.
- WithdrawalContract: The WithdrawalContract library is a secure way to manage and distribute funds in a decentralized manner. It allows for the safe storage and withdrawal of ether and other ERC20 tokens, which could be useful in a crowdfunding platform that accepts multiple types of tokens.

3.5.3 VS Code - IDE

VS Code (Visual Studio Code) is a free and open-source code editor developed by Microsoft. VS Code is available for Windows, Linux and macOS. Although the editor is relatively lightweight, it includes some powerful features that have made VS Code one of the most popular development environment tools in recent times. Moreover, VS Code allows you to add on and even creating new extensions including code linters, debuggers and cloud and web development support. The VS Code user interface allows for a lot of interaction compared to other text editors. To simplify user experience, VS Code is divided into five main regions such as the activity bar, the side bar, editor groups, the panel, the status bar.

Chapter 4

SYSTEM TESTING

Testing is the process of detecting errors, performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing play a key role in maintenance phase also. The aim of testing is often to demonstrate that a program works by showing that it has no errors. The basic purpose of testing phase is to detect the errors that may be present in the program. The intent of testing should be to show that a program doesn't work. The various testing performed are unit testing, integration testing, validation testing, output testing and system testing.

4.1 Types of Test Cases

- Integration Test case: An integration test case is a type of software testing that focuses on testing the interactions between different components or modules of a software system. Integration testing is typically performed after unit testing and before system testing, and it involves testing the integration between different components to ensure that they work together seamlessly. Integration test cases are designed to validate the interactions between different components and identify any issues that might arise when the components are combined.

- Functional Test case: It is focuses on testing the functionality of a software system. The purpose of functional testing is to ensure that the software system meets the requirements and specifications defined for it. Functional test cases are typically written based on the requirements or user stories and are designed to test the system's features, user interactions, data processing, and output.

- Non-Functional Test Case: It is a type of software testing that focuses on testing the non-functional aspects of a software system, such as performance, usability, security,reliability, and scalability. Non-functional test cases are typically written to validate the system's behaviour under different non-functional requirements and constraints, such as load, stress, concurrency, and availability.

- User Acceptance Test case: These test cases are crucial and very important to client-side peoples, because these test case talks about these business and approach of the application to complete a particular client task, which is also called as End-End Business scenario test case. Here we won't be doing testing relates to UI, Functionality or Non-Functionality, we talk about business and scenario which the application is made for.

4.2 Test Case Design And Execution

The test case is a document that describes an input, action, or event and an expected response to determine if a feature of an application is working correctly. A test case should contain particulars such as test case identifiers, test case, name, objectives, test conditions, input data requirements steps and expected results.

4.2.1 Test Case Of Campaign Creation

Test Scenario ID	Campaign Creation	**Pre-Requisite**	Wallet Connection
Test case description	Testing creation of campaign	**Post- Requisite**	NA

Sl No.	Action	Inputs	Result
1.	Filling all the fields	Entering values in each field matching constraints	Allowing user to click **Submit Project** button
2.	Filling all the fields	Missing few input fields	Showing validation error.
3.	Filling all the fields	Entered all the fields but mismatching constraints of the field	Showing validation error.
4.	Authorizing creation	Insufficient funds in wallet to deploy the smart contract.	Report **Insufficient funds** error.
5.	Authorizing creation	Sufficient funds in wallet to deploy the smart contract.	Authorizing with wallet.

Figure 3.1: Test case of campaign creation

4.2.2 Test Case Of Contributing To Campaign

Test Scenario ID	Contributing to campaign	**Pre-Requisite**	Wallet Connection
Test case description	Testing contribution	**Post- Requisite**	NA

Sl No.	Action	Inputs	Result
1.	Contributing funds	Entering < minimum amount & Clicking Contribute Funds button.	Showing validation error.
2.	Contributing funds	Entering >= minimum amount & Clicking Contribute Funds button.	Allowing user to authenticate the transaction with wallet.
3.	Contributing funds	Entering >= target amount & Clicking Contribute Funds button	Allowing user to authenticate the transaction with wallet.

Figure 3.2: Test case of campaign creation

4.2.3 Test Case Of Campaign Aborting

Test Scenario ID	Campaign Aborting	**Pre-Requisite**	Connected with fundraiser's wallet
Test case description	Testing aborting	**Post- Requisite**	NA

Sl No.	Action	Inputs	Result
1.	Aborting campaign	Clicking **Delete** button (before deadline)	Showing wallet to authorize transaction
2.	Aborting campaign	Clicking **Delete** button (before refresh goal reach)	Showing wallet to authorize transaction

Figure 3.3: Test case of campaign aborting

4.2.4 Test Case Of Display Campaign

Test Scenario ID	Displaying Campaign	**Pre-Requisite**	Connected with fundraiser's wallet
Test case description	Testing displaying	**Post- Requisite**	NA

Sl No.	**Action**	**Inputs**	**Result**
1.	Clicking on campaign card	Clicking on a campaign card in home page	Showing respective campaign page

Figure 3.4: Test case of Display campaign

Chapter 5

RESULT AND DISCUSSION

The utilization of blockchain technology in crowdfunding projects brings forth several notable results. Firstly, it enhances transparency by leveraging the immutable nature of blockchain, ensuring that all transactions and activities related to the crowdfunding project are recorded in a transparent and verifiable manner. This transparency instills trust among participants, as they can track the flow of funds and ensure their proper allocation.

Secondly, blockchain provides a robust security framework for crowdfunding endeavors. With its cryptographic techniques, blockchain safeguards transactions against tampering and unauthorized access. This heightened security not only protects the funds contributed by backers but also reduces the risks associated with fraudulent activities, bolstering confidence in the crowdfunding ecosystem. Furthermore, blockchain-powered crowdfunding projects can lower costs by eliminating or minimizing the need for intermediaries like banks or payment processors. By utilizing blockchain's decentralized nature, these projects can streamline transactions and reduce fees, allowing more funds to directly reach the project and its intended beneficiaries.

5.1 Results

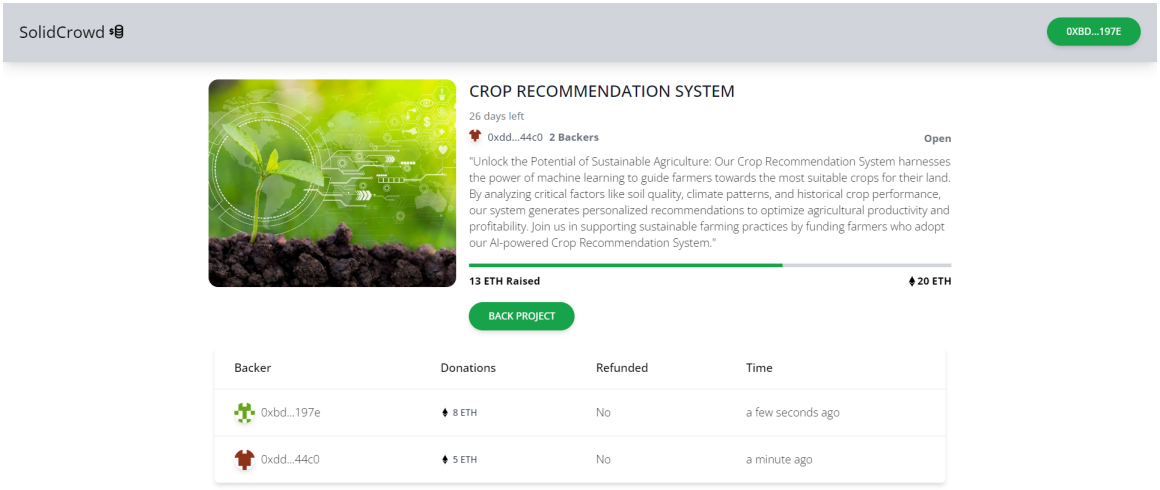

Figure 3.1: Investors Backing The Project

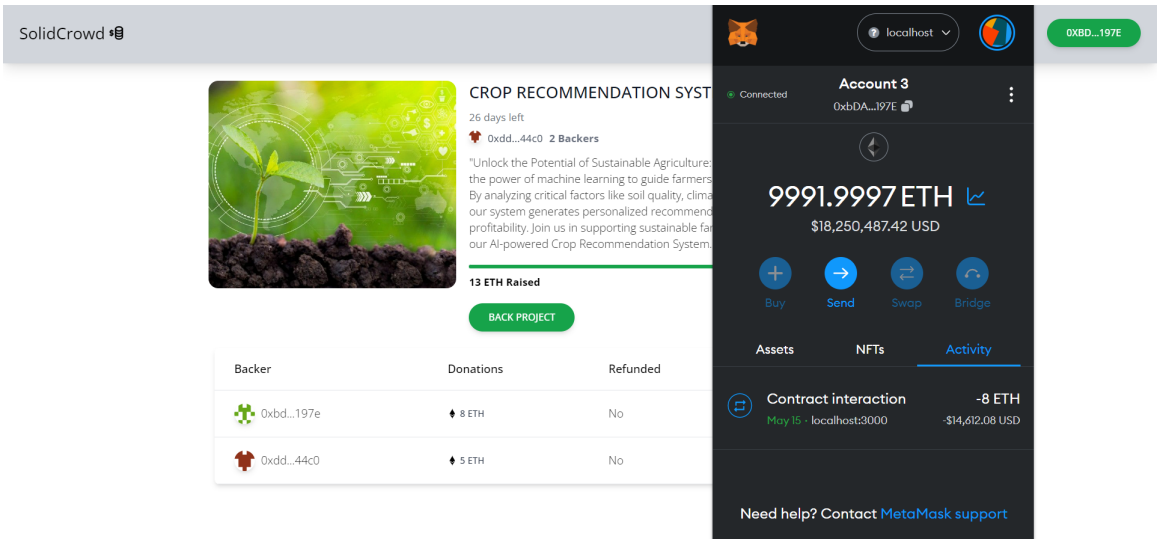

Figure 3.2: Contribution of 8 ETH from investor

Chapter 6

CONCLUSION

As the world is moving towards Web 3.0 and decentralized systems to solve their daily problems, it is important to test and build new alternative architectures that show us the ideology to provide innovative solutions. With the existing solutions in the crowdfunding world created and handled by intermediary corporations that have a say on various parameters of a campaign, the alternative solution based on peer-to-peer network handling the campaign transactions seems ripe. This project explores ways to remove intermediaries in a crowdfunding business use case. This was done with the help of smart contracts, written for the crowdfunding dapp application deployed in Ethereum blockchain, that guide the execution of a transaction. This interaction allows users to create and invest ether into campaigns that interest them. Without much efforts campaign creators and campaign investors can perform their intended activities using the crowdfunding platform. There are new emerging blockchain platforms such as EOS, Stellar, Cardano and NEO [that provide more language choices and platform configuration choices compared to Ethereum but these platforms haven't proved themselves yet. EOS looks like a promising platform and in future this project can be moved to EOS if it proves to be a better choice than Ethereum. In order to conclude we would like to affirm that the potential and influence that the emerging technologies possess, for crowdfunding, is immense. The current solutions for the challenges that the usual crowdfunding platforms pose, are now able to transform the society for better. Crowdfunding platforms using the blockchain technology hold more credibility and therefore, we believe are the future for the right investment for investors.

6.1 Future Enhancement

In the future, there are several potential enhancements that could be implemented to further improve the crowdfunding project using blockchain. One possible enhancement is the integration of decentralized identity (DID) solutions, such as Self-Sovereign Identity (SSI) frameworks. This would provide a secure and privacy-preserving way to verify the identity of project creators and backers, increasing trust and preventing identity fraud within the platform.

Another enhancement could involve the integration of reputation systems. By implementing mechanisms such as token-based reputation or user ratings, backers would be able to make more informed decisions based on the reputation and track record of project creators. This would further enhance trust and credibility within the crowdfunding platform.

Additionally, exploring partnerships with external service providers, integrating with other blockchain networks or solutions, and optimizing scalability and performance are essential for future enhancements. These actions would ensure regulatory compliance, unlock additional functionalities, and improve the overall efficiency and user experience of the crowdfunding platform.

REFERENCES

[1] Muneeza, Aishath, Nur Aishah Arshad, and Asma Tajul Arifin. *"The application of blockchain technology in crowdfunding: towards financial inclusion via technology." International journal of management and applied research 5, no. 2 (2018): 82-98.*

[2] Pandey, S., Goel, S., Bansla, S. and Pandey, D., 2019, March. *Crowdfunding fraud prevention using blockchain. In 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 1028-1034). IEEE.*

[3] Yadav, N. and Sarasvathi, V., 2020, August. *Venturing crowdfunding using smart contracts in blockchain. In 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 192-197). IEEE.*

[4] Gururaj, H.L., Janhavi, V., Holla, A.M., Kumar, A.A., Bhumika, R. and Goundar, S., 2021. *Decentralised application for crowdfunding using blockchain technology. International Journal of Blockchains and Cryptocurrencies, 2(1), pp.68-82.*

[5] Bogusz, C.I., Laurell, C. and Sandström, C., 2020. *Tracking the digital evolution of entrepreneurial finance: the interplay between crowdfunding, blockchain technologies, cryptocurrencies, and initial coin offerings. IEEE Transactions on Engineering Management, 67(4), pp.1099-1108.*

[6] Basu, S., Basu, K. and Austin, T.H., 2022, February. *Crowdfunding Non-fungible Tokens on the Blockchain. In Silicon Valley Cybersecurity Conference: Second Conference, SVCC 2021, San Jose, CA, USA, December 2–3, 2021, Revised Selected Papers (pp. 109-125). Cham: Springer International Publishing.*

[7] Khatter, H., Chauhan, H., Trivedi, I. and Agarwal, J., 2021, August. *Secure and transparent crowdfunding using blockchain. In 2021 international conference on recent*

trends on electronics, information, communication and technology (RTEICT) (pp. 76-80). IEEE.

[8] Berawi, M.A., Radjilun, M.K.Z. and Sari, M., 2021, August. *Developing Blockchain-Based Crowdfunding Model for Property Investment. In Innovations in Digital Economy: Second International Scientific Conference, SPBPU IDE 2020, St. Petersburg, Russia, October 22–23, 2020, Revised Selected Papers (pp. 23-39). Cham: Springer International Publishing.*

[9] Bracamonte, V. and Okada, H., 2017. *An exploratory study on the influence of guidelines on crowdfunding projects in the ethereum blockchain platform. In Social Informatics: 9th International Conference, SocInfo 2017, Oxford, UK, September 13-15, 2017, Proceedings, Part II 9 (pp. 347-354). Springer International Publishing.*

APPENDIX

Screenshots

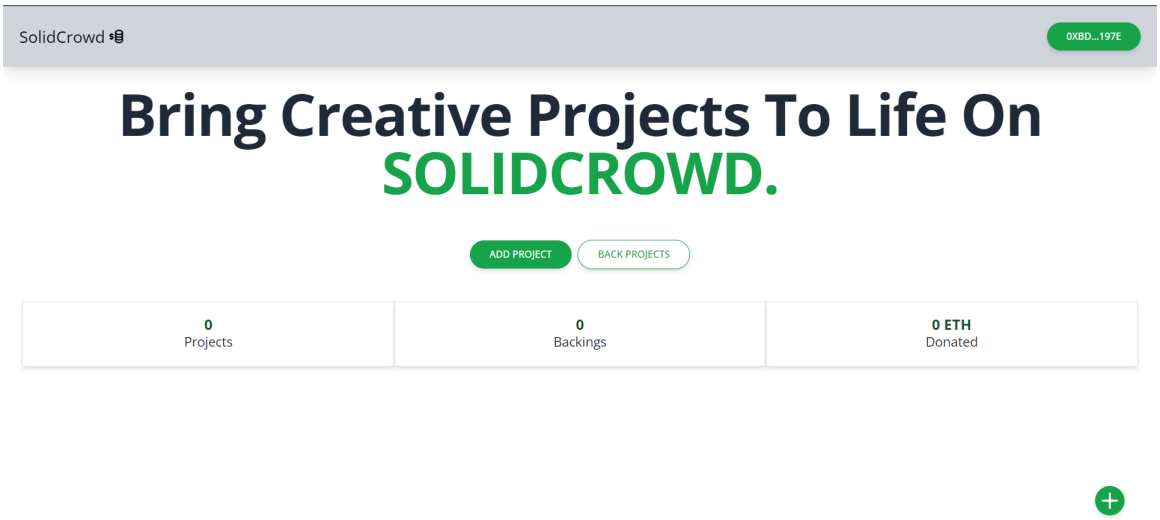

Figure A.1: Homepage

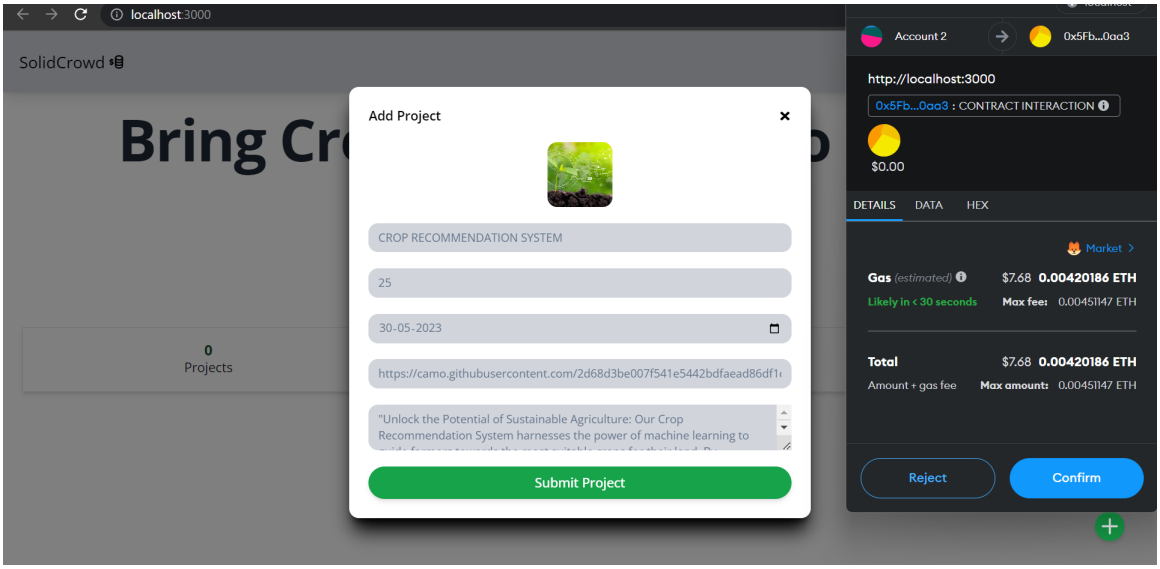

Figure A.2: Creating a Campaign

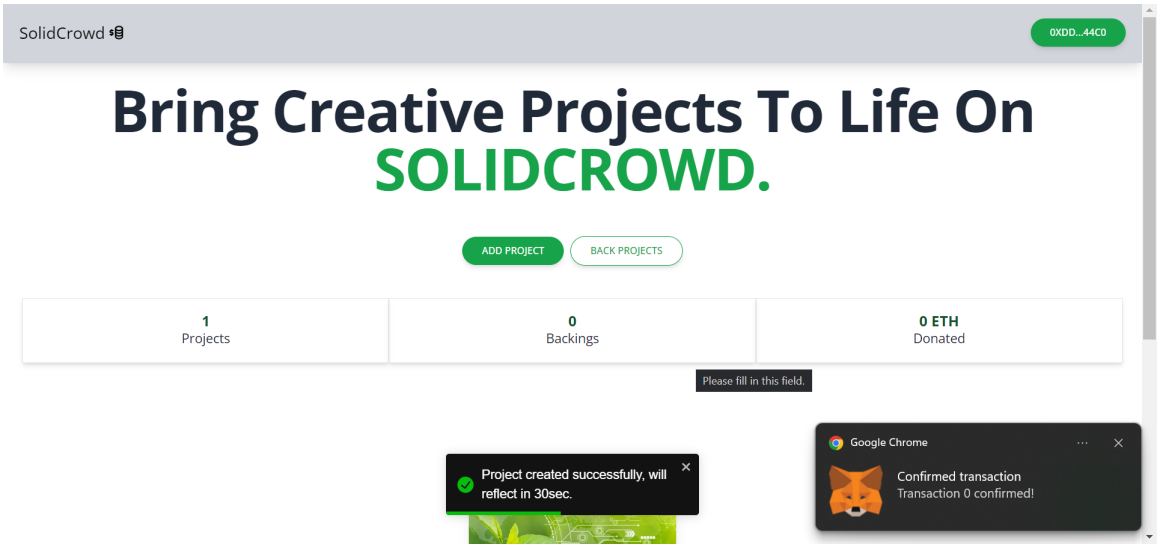

Figure A.3: After transaction successful campaign list to Homepage

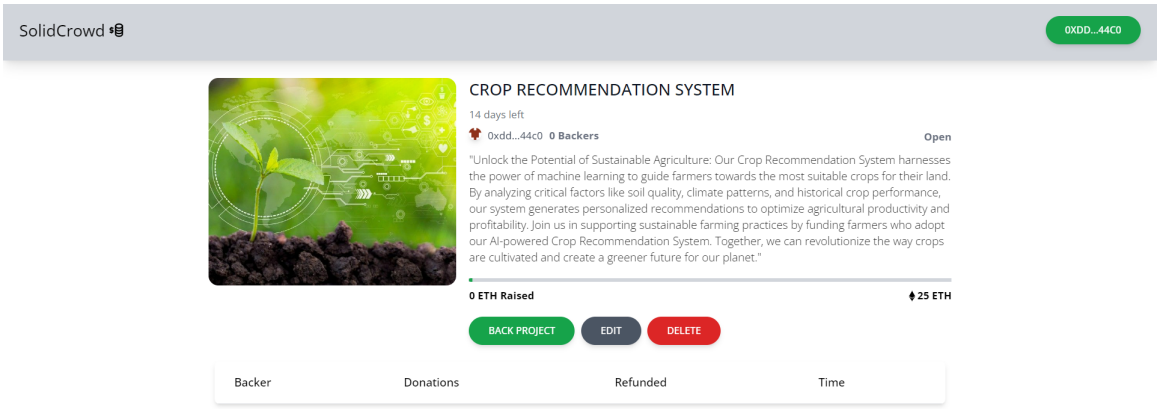

Figure A.4: Campaign Dashboard (Creator view)

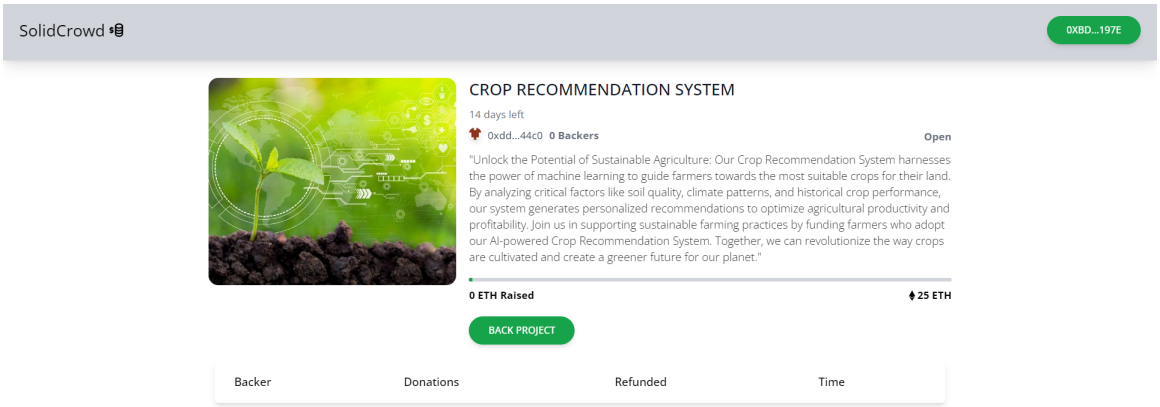

Figure A.5: Campaign Dashboard (Investor's View)

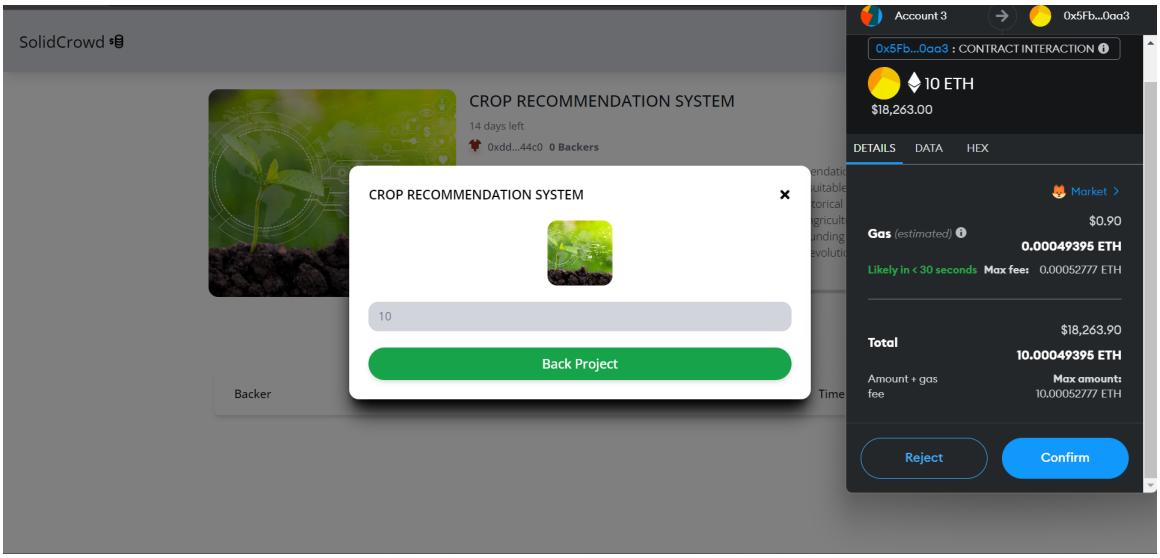

Figure A.6: Investor backing the campaign and MetaMask Transaction popup

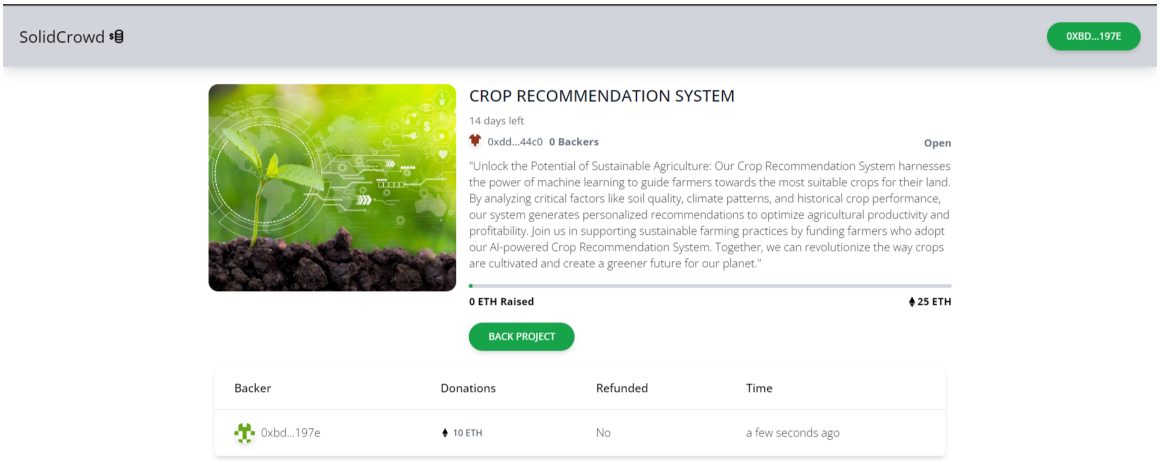

Figure A.7: Added Raised amount to Campaign page (Campaign Open).

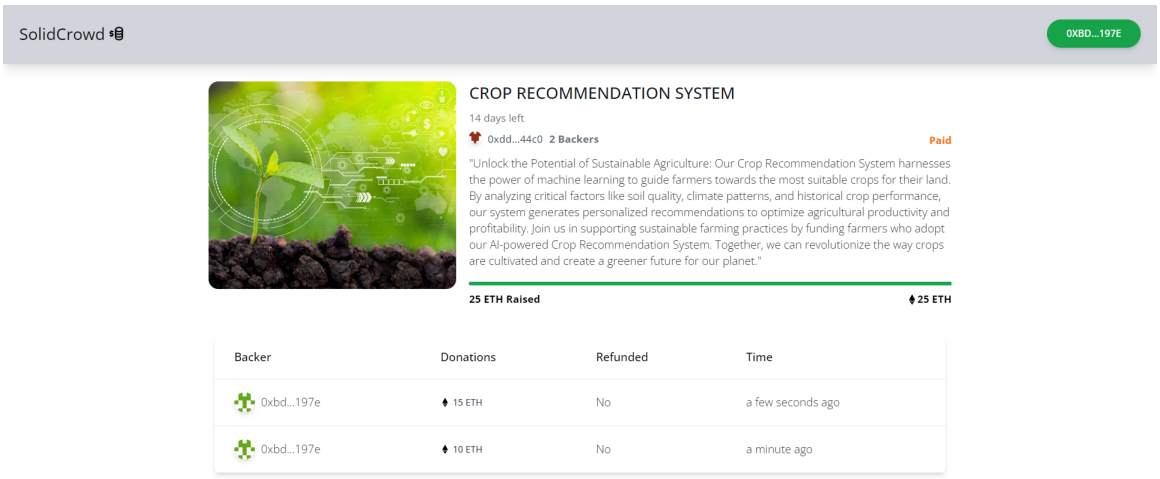

Figure A.8: Campaign Dashboard (After Campaign closed).

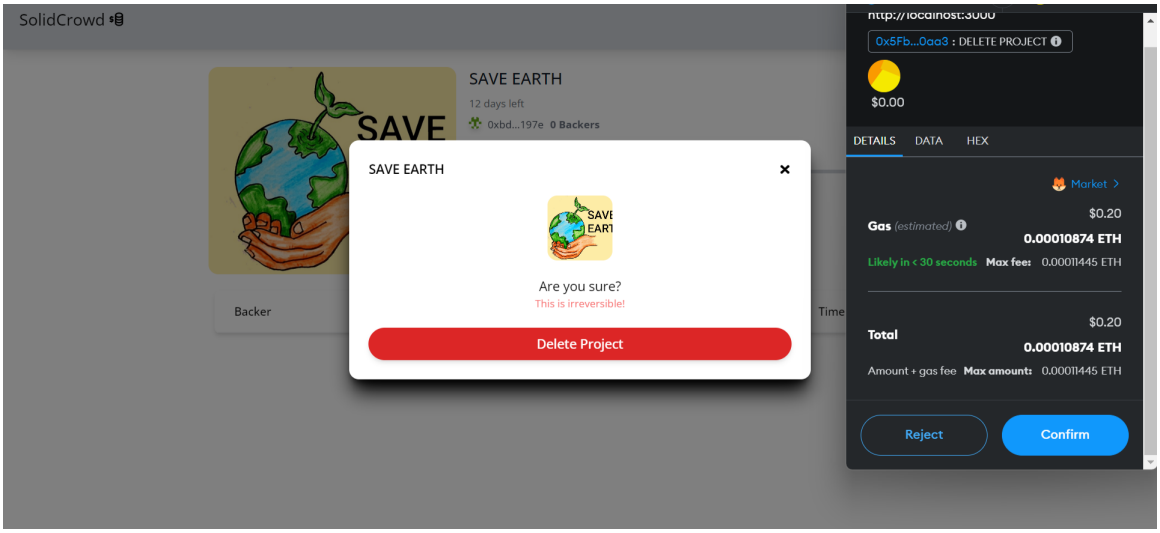

Figure A.9: Campaign creator deleting the Campaign.